

SYSTEM FOR OUTPUTTING ERROR MESSAGE OF COMPILER

Patent Number: JP6044081
Publication date: 1994-02-18
Inventor(s): YABE YUICHI
Applicant(s):: HOKKAIDO NIPPON DENKI SOFTWARE KK
Requested Patent: ☐ JP6044081
Application Number: JP19910297463 19911113
Priority Number(s):
IPC Classification: G06F9/45 ; G06F9/06 ; G06F11/28
EC Classification:
Equivalents:

Abstract

PURPOSE:To aid in resolving a compile error at an early stage when a program is developed.
CONSTITUTION:A syntax analyzing means 2 input an inputs source program which is described in a PASCAL language and analyzes syntax in accordance with the syntax rule of the PASCAL language. A symbol table registering means 3 registers information of a variable name and a type name in a symbol table 8. At the time of the error which is detected by the noncoincidence, etc., of the variable name and the type name, an index registered in the symbol table 8 as the corresponding variable name and the type name, information of an error table 10, information of an error message table and information of the symbol table 8 are edited and an error message outputting means 6 outputs a result to an output list 7.

Data supplied from the **esp@cenet** database - I2

(19) 日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平 6 - 4 4 0 8 1

(43) 公開日 平成6年(1994)2月18日

(51) Int. Cl. ⁵	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F	9/45			
	9/06	4 4 0 S	9367 - 5 B	
	11/28	E	9290 - 5 B	
			9292 - 5 B	
			G 0 6 F	9/44 3 2 2 C

審査請求 未請求 請求項の数 3

(全 6 頁)

(21) 出願番号 特願平3-297463

(22) 出願日 平成3年(1991)11月13日

(71) 出願人 000241979

北海道日本電気ソフトウェア株式会社

北海道札幌市中央区南一条西4丁目5番地1号

(72) 発明者 矢部 祐一

北海道札幌市中央区南一条西四丁目5番地1号 北海道日本電気ソフトウェア株式会社内

(74) 代理人 弁理士 後藤 洋介 (外2名)

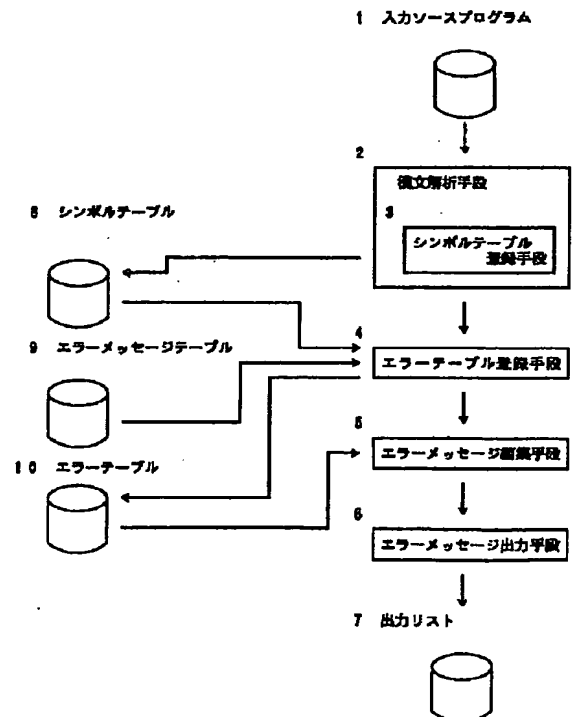
請求項1 (X)

(54) 【発明の名称】 コンパイラのエラーメッセージ出力方式

(57) 【要約】

【目的】 プログラム開発時においてコンパイルエラーの早期解決に役立てる。

【構成】 構文解析手段2は、PASCAL言語で記述された入力ソースプログラム1を入力し、PASCAL言語の構文規則に従い構文解析を行う。シンボルテーブル登録手段3は変数名および型名の情報をシンボルテーブル8に登録する。変数名および型名の不一致等により検出したエラーの場合は対応する変数名または型名の情報としてシンボルテーブル8に登録されているインデックスをエラーテーブル10の情報とエラーメッセージテーブル9の情報を編集し、エラーメッセージ出力手段6により、出力リスト7に結果を出力する。



【特許請求の範囲】

【請求項1】 PASCAL言語で記述されたソースプログラムのコンパイル時に出力されるエラーメッセージ出力方式において、
前記ソースプログラムの構文を分析して、前記PASCAL言語の構文規則とは異なる場合には、コンパイルエラーとして認識する構文分析手段と、
該構文分析手段により解析された情報をシンボルテーブルに登録するシンボルテーブル登録手段と、
前記コンパイルエラーに関するエラー情報をエラーテーブルに登録するエラーテーブル登録手段と前記コンパイルエラーに対応するエラーメッセージを予め格納するエラーメッセージテーブルと、
前記シンボルテーブルに登録されている情報を引用すると共に、前記エラーテーブルに登録されている前記コンパイルエラーに対応したエラーメッセージを、前記エラーメッセージテーブルから引用して、所定の形式に編集するエラーメッセージ編集手段とを有することを特徴とするコンパイラのエラーメッセージ出力方式。

【請求項2】 請求項1記載のコンパイラのエラーメッセージ出力方式において、前記シンボルテーブルに登録される情報は、前記ソースプログラムの変数名および型名を含むことを特徴とするコンパイラのエラーメッセージ出力方式。

【請求項3】 請求項2記載のコンパイラのエラーメッセージ出力方式において、前記エラー情報は、コンパイルエラーのあったライン数と、カラム位置とコンパイルエラーの番号とを含み、前記ソースプログラムの変数名および型名に関連したコンパイルエラーの場合は、前記シンボルテーブルに規定されるインデックス番号を含むことを特徴とするコンパイラのエラーメッセージ出力方式。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、PASCAL言語で記述されたソースプログラムの構文解析時にエラーが検出された場合、変数名及び型名の不一致等からエラーが検出された場合は、関連する変数名と型名およびソース行とをエラーメッセージとともに出力することを可能とする、コンパイラのエラーメッセージ出力方式に関するものである。

【0002】

【従来の技術】従来、PASCAL言語で記述されたソースプログラムをコンパイルした結果、検出されたコンパイルエラーを出力する場合には、エラーのあった行と、その行のどこの位置でエラーが発見されたかを示す記号と、エラーの種類を示すエラー番号と、エラーメッセージを1つの単位としてエラーメッセージを出力している。

【0003】

【発明が解決しようとする課題】しかしながら、従来、PASCAL言語で記述されたソースプログラムをコンパイルした結果、検出したコンパイルエラーを出力する場合には、エラーのあった行と、その行のどこの位置でエラーが発見されたかを示す記号と、エラーの種類を示すエラー番号と、エラーメッセージを1つの単位としてエラーメッセージを出力しているため、1つのコンパイルエラーを解決するためには、変数及び型の宣言部を参照しなければならないといった作業が生じる欠点がある。そこで、本発明の技術的課題は、上記欠点に鑑み、コンパイルエラーを検出した位置に、変数及び型の不一致等に関するエラーであればエラーメッセージとともにその変数名と型名とそのソース行を表示することができるため、コンパイラのエラーメッセージ出力方式を提供することである。

【0004】

【課題を解決するための手段】本発明によれば、PASCAL言語で記述されたソースプログラムのコンパイル時に出力されるエラーメッセージ出力方式において、前記ソースプログラムの構文を分析して、前記PASCAL言語の構文規則とは異なる場合には、コンパイルエラーとして認識する構文分析手段と、該構文分析手段により解析された情報をシンボルテーブルに登録するシンボルテーブル登録手段と、前記コンパイルエラーに関するエラー情報をエラーテーブルに登録するエラーテーブル登録手段と前記コンパイルエラーに対応するエラーメッセージを予め格納するエラーメッセージテーブルと、前記シンボルテーブルに登録されている情報を引用すると共に、前記エラーテーブルに登録されている前記コンパイルエラーに対応したエラーメッセージを、前記エラーメッセージテーブルから引用して、所定の形式に編集するエラーメッセージ編集手段とを有することを特徴とするコンパイラのエラーメッセージ出力方式が得られる。

【0005】また、本発明によれば、前記のコンパイラのエラーメッセージ出力方式において、前記シンボルテーブルに登録される情報は、前記ソースプログラムの変数名および型名を含むことを特徴とするコンパイラのエラーメッセージ出力方式が得られる。

【0006】また、本発明によれば、前記のコンパイラのエラーメッセージ出力方式において、前記エラー情報は、コンパイルエラーのあったライン数と、カラム位置とコンパイルエラーの番号とを含み、前記ソースプログラムの変数名および型名に関連したコンパイルエラーの場合は、前記シンボルテーブルに規定されるインデックス番号を含むことを特徴とするコンパイラのエラーメッセージ出力方式が得られる。

【0007】即ち、本発明のコンパイラのエラーメッセージ出力方式では、入力されたソースプログラム構文を解析する構文解析手段と、ソースプログラム中の宣言部で宣言された変数名及び型名等の情報を作成するシンボ

ルテーブル登録手段と、シンボルテーブル登録手段により作成されるシンボルテーブルと、コンパイラのエラーメッセージを含むエラーメッセージテーブルと、コンパイラの構文解析手段によって検出したエラーとシンボルテーブル登録手段により登録されたシンボルテーブルをエラーテーブルに登録するエラーテーブル登録手段と、エラーテーブル登録手段により情報が登録されるエラーテーブルと、エラーテーブル登録手段により登録されたエラーテーブルの内容を編集するエラーメッセージ編集手段と、エラーメッセージ編集手段により編集されたエラーメッセージを出力するエラーメッセージ出力手段とを有している。

【0008】

【実施例】次に本発明の実施例について図面を参照して詳細に説明する。

【0009】図1は本実施例によるコンパイラのエラーメッセージ出力方式を実施する一実施例を示すブロック図である。

【0010】同図においてコンパイラのエラーメッセージ出力方式は、入力ソースプログラム1、構文解析手段2、シンボルテーブル登録手段3、エラーテーブル登録手段4、エラーメッセージ編集手段5、エラーメッセージ出力手段6、出力リスト7、シンボルテーブル8、エラーメッセージテーブル9、エラーテーブル10、から構成されている。

【0011】構文解析手段2は、PASCAL言語で記述された入力ソースプログラム1を入力し、PASCAL言語の構文規則に従い構文解析を行い、構文規則に従わない場合は、コンパイルエラーとして認識し、コンパイルエラーがあった入力ソースプログラム1のソース行とカラム位置とコンパイルエラーの番号を出力する。

【0012】またシンボルテーブル登録手段3は、変数名と型名およびそのソース行をシンボルテーブル8に登録する。

【0013】変数名および型名の不一致等により検出したエラーの場合は、対応する変数名または型名の情報としてシンボルテーブル8に登録されているインデックスをエラーテーブル登録手段4によりエラーテーブル10に登録する。

【0014】エラーメッセージ編集手段5は、エラーテーブル10に登録されている情報とコンパイルエラーの番号に対応したエラーメッセージをエラーメッセージテーブル9から引用し、さらに変数名および型名に対応した情報をシンボルテーブル8から引用し、その結果を編集する。

【0015】編集された結果はエラーメッセージ出力手段6により、出力リスト7に結果を出力する。

【0016】図2は、入力ソースプログラムの例11である。

【0017】図3は、コンパイルエラーが検出された場

合に出力されるコンパイルエラーメッセージが格納されている、エラーメッセージテーブル12である。

【0018】図4は、変数名および型名の情報が格納されているシンボルテーブル13である。

【0019】図5は、構文解析手段2によって生成された情報を、エラーテーブル登録手段4によってエラーテーブル10に登録した結果生成された、エラーテーブル10の内部形式の説明図である。

【0020】構文解析手段2によって生成されたコンパイルエラーのあったライン数とカラム位置とコンパイルエラーの番号を、それぞれエラーテーブル14のlineとcolumnとエラーメッセージNoというフィールドに登録する。

【0021】さらに変数名および型名に関連したコンパイルエラーの場合は、エラーテーブル14のシンボルテーブルNoというフィールドにシンボルテーブル8のインデックス番号を登録する。

【0022】図6は、入力ソースプログラムの例11を構文解析手段2に入力し、エラーテーブル登録手段4とエラーメッセージ編集手段5とエラーメッセージ出力手段6で処理した結果出力される、出力リストの例15である。

【0023】

【発明の効果】以上説明したように、本発明では、PASCAL言語で記述されたソースプログラムに対して、コンパイルエラーを検出した位置に、変数名および型名に関するエラーであればエラーメッセージとともに変数名と型名およびそのソース行の情報を表示することができ、ソースプログラムを開発し、コンパイルした際に検出されたコンパイルエラーに対してソースプログラムの修正が行い易くなるという利点がある。

【図面の簡単な説明】

【図1】本発明の全体構成図を示す図である。

【図2】PASCAL言語で記述されたプログラム例である。

【図3】エラーメッセージテーブルの説明図である。

【図4】シンボルテーブルの説明図である。

【図5】エラーテーブルの説明図である。

【図6】実行の結果出力される出力結果の例である。

【符号の説明】

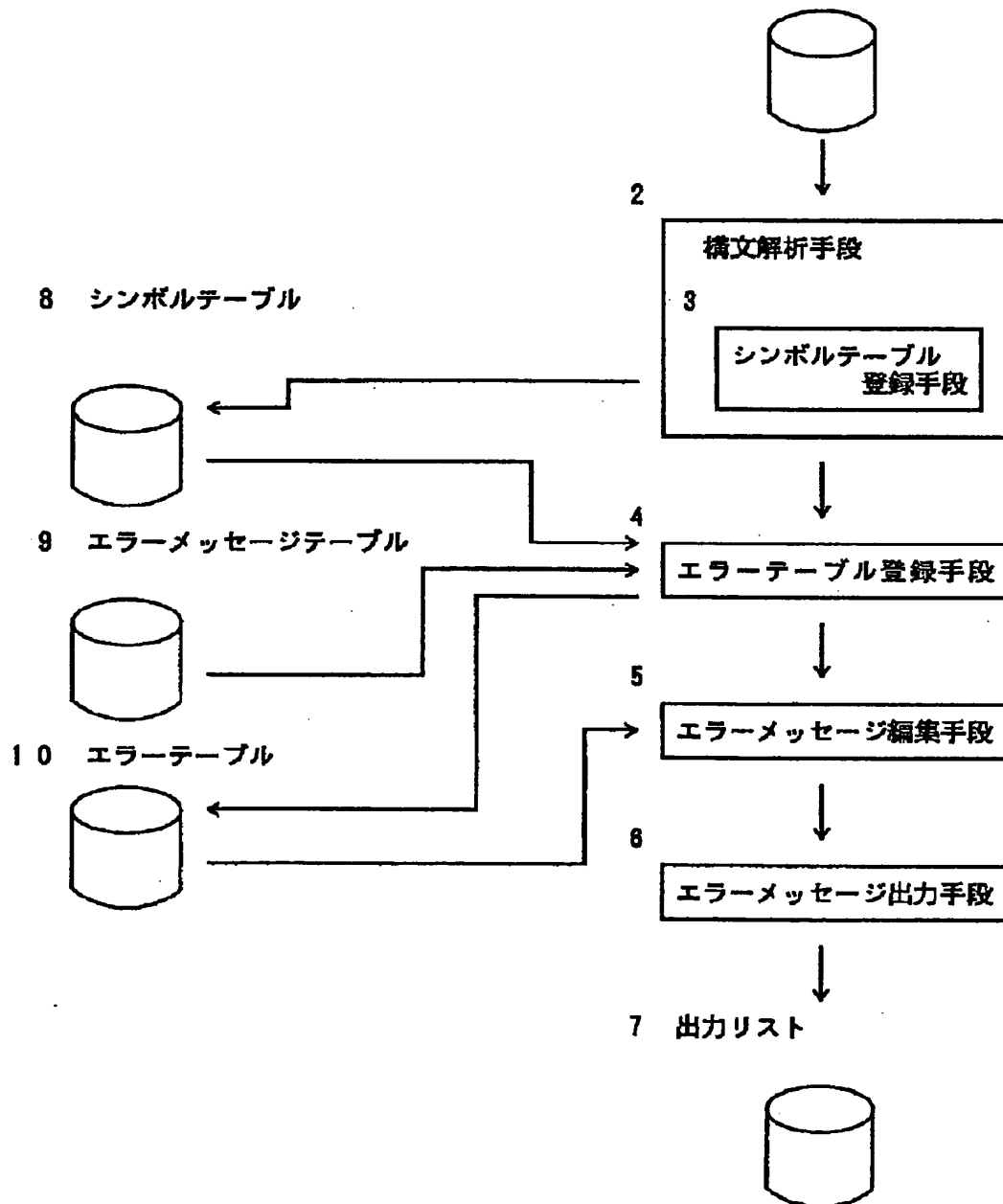
- 1 入力ソースプログラム
- 2 構文解析手段
- 3 シンボルテーブル登録手段
- 4 エラーテーブル登録手段
- 5 エラーメッセージ編集手段
- 6 エラーメッセージ出力手段
- 7 出力リスト
- 8 シンボルテーブル
- 9 エラーメッセージテーブル
- 10 エラーテーブル

5
 1 1 入力ソースプログラムの例
 1 2 エラーメッセージテーブル
 1 3 シンボルテーブル

6
 1 4 エラーテーブル
 1 5 出力リストの例

【図1】

1 入力ソースプログラム



【図2】

1.1 入力ソースプログラムの例

LINE	123456789 * 123456789 * 123456789 * 123456789 *
01	program sample ;
02	var
03	i : integer ;
04	b : boolean ;
05	r: real;
06	begin
07	readln(r);
08	if r then
09	i := r;
10	new(b);
11	end.

【図3】

1.2 エラーメッセージテーブル

No.	エラーメッセージ
1	begin が必要
2	end が必要
3	; が必要
4	演算子が必要
5	代入文の左辺と右辺の型が異なる
6	レコード型の変数でなければならない
7	論理型の式でなければならない
8	整数型の変数に実数値は代入できない
9	for 文の制御変数には実数型の変数を指定できない
10	ポインタ型の変数でなければならない

【図4】

【図5】

1.3 シンボルテーブル

No.	変数名	型名	ソース行
1	i	integer	3
2	b	boolean	4
3	r	real	5

1.4 エラーテーブル

No.	line	column	エラーメッセージNo.	シンボルテーブルNo.
1	8	6	7	3
2	9	10	5	1, 3
3	10	7	10	2

【図6】

15 出力リストの例

```
01 program sample;
02 var
03   i : integer ;
04   b : boolean ;
05   r : real ;
06 begin
07   readln(r) ;
08   if r then
      ^7
      **7 論理型の式でなければならない
      ** r は real          5 行目

09   i := r ;
      ^5
      **5 代入文の左辺と右辺の型が異なる
      ** i は integer      3 行目
      ** r は real         5 行目

10   new(b) ;
      10
      **10 ポインタ型の変数でなければならない
      ** b は boolean      4 行目

11 end.
```